ACEEE Int. J. on Information Technology, Vol. 3, No. 2, June 2013

Multi Similarity Measure based Result Merging Strategies in Meta Search Engine

K.Srinivas[1], V.Valli Kumari[2], and A.Govardhan[3]
[1] Geethanjali College of Engineering & Technology, Department of Information Technology, Andhra Pradesh, India
katkamsrinu@gmail.com
[2]Professor, Department of CS & SE, Andhra University College of Engineering, Andhra University, Visakhapatnam,
valli_kumari@rediffmail.com
[3]Professor, Department of CSE, Jawaharlal Nehru Technological University, Hyderabad, Andhra Pradesh, India
govardhan_cse@yahoo.co.in

Abstract—**In Meta Search Engine result merging is the key component. Meta Search Engines provide a uniform query interface for Internet users to search for information. Depending on users' needs, they select relevant sources and map user queries into the target search engines, subsequently merging the results. The effectiveness of a Meta Search Engine is closely related to the result merging algorithm it employs. In this paper, we have proposed a Meta Search Engine, which has two distinct steps (1) searching through surface and deep search engine, and (2) Ranking the results through the designed ranking algorithm. Initially, the query given by the user is inputted to the deep and surface search engine. The proposed method used two distinct algorithms for ranking the search results, concept similarity based method and cosine similarity based method. Once the results from various search engines are ranked, the proposed Meta Search Engine merges them into a single ranked list. Finally, the experimentation will be done to prove the efficiency of the proposed visible and invisible web-based Meta Search Engine in merging the relevant pages. TSAP is used as the evaluation criteria and the algorithms are evaluated based on these criteria.**

Index Terms — **Meta search engine, ranking, concept, cosine similarity, deep web, surface web**.

I. Introduction

Meta Search Engines provide a uniform query interface for Internet users to search for information. Depending on users needs, they select relevant sources and map user queries into the target search engines, subsequently merging the results. However, considering the great diversity in schematic, semantic, interface, and domain aspects, it is very important but quite difficult to make full use of the functions of specific search engines. Furthermore, in the educational context, the massification of the Web and search engines, has contributed to access large bibliographic contents, much larger than the generally needed for their assignments [4]. A Meta Search Engines provides a single integrated interface, where a user enters an specific query, the engine forwards it in parallel to a given list of search engines, and results are collated and ranked into a single list [4,8]. Meta Search Engines do not crawl the Internet themselves to build an index of Web documents. Instead, a Meta Search Engine sends queries simultaneously to multiple other Web search engines, retrieves the results from each, and then combines the results from all into a single result, at the same time avoiding redundancy. In effect, Web Meta Search Engine users are not using just one engine, but many search engines at once to effectively employ Web searching [9]. Although one could certainly query multiple search engines, a Meta Search Engine purifies these top results automatically, giving the searcher a comprehensive set of search results within a single listing, all in real time [9].

Many people use search engines to find their requirements on the web. Researches show that each search engines covers some parts of the web. Therefore, Meta Search Engines are invented to combine results of different search engines and increase web search effectiveness due to a larger coverage of indexed web. Today's Meta Search Engine's activities are more than a simple combination of search engine results. They try to create profiles for their users and personalize search results by taking these profiles into account. This process is called Search Personalization and its usage is not limited to Meta Search Engines [1]. Many Meta Search Engines are created for the purpose of combining results of different information retrieval systems such as Profusion [10], SaavySearch [11], WebFusion [13], I-Spy [14], a few to name. Some of them use Multi-agent systems for their architecture [10].

Chignell et al. [15] found little overlap in the results returned by various Web search engines. They describe a Meta Search Engine as useful, since different engines employ different means of matching queries to relevant items, and also have different indexing coverage. Selberg et al. [16] further suggested that no single search engine is likely to return more than 45% of the relevant results. Subsequently, the design and performance of Meta Search Engines have become an ongoing area of study. The search engines admits a fixed number of characters in their queries, for which the document needs to be chopped up into several parts, and then delivered in parts to the search engine [4]. Thus a solution has been relevant to alter the current status of the Meta Search Engines. Our proposed method is keeping an eye on the improvement of the search criteria of the Meta Search Engines.

In reference to the above stated problems, we tried to develop an advanced Meta Search Engine. The process of

DOI: 01.IJIT.3.2.1194

the proposed visible and invisible web based Meta Search Engine is divided into two major steps, (1) searching through surface and deep search engine, and (2) Ranking the results through the designed ranking algorithm. Initially, the query given by the user is inputted to the deep and surface search engine. Here, the surface search engine like Google, bing and yahoo can be considered. At the same time, the deep search engine such as, Infomine, Incywincy and Complete Planet can be considered. Once more number of pages obtained from the visible and invisible web, the ranking of those pages should be carried out to provide the most relevant pages. The ranking of those pages will be carried out using the proposed algorithm that considers the similarity of input query to those web pages as well as the inter-similarity among the web pages retrieved. In inter-similarity of web pages, the concept based similarity measure will be used. Finally, the experimentation will be done to prove the efficiency of the proposed visible and invisible web based Meta Search Engine in merging the relevant pages [12].

The main contributions of our proposed approach are the two distinct algorithms that we have adapted for the search in the web. The algorithms are based on similarity measures, one algorithm is based on concept similarity and other is based on cosine similarity. We use the search results from both surface web search and deep web search as the input to the two algorithms. The proposed approach has given significant result at the experimentation phase.

The rest of the paper is organized as, the section 2 gives a review of some related works regarding web search and Meta Search Engines. Section 3 contains Motivational algorithms behind this research. Section 4 gives details of the proposed method with mathematical models. 5th section gives the results and discussion about the proposed method and with the 6th section we conclude our research work.

II. Related Works

In this section, we have plotted some of the latest researches regarding the Meta Search algorithms. The most of the researches are trending towards optimizing search process. Most of the results concentrate on the improvement of the efficiency of the Meta Search results.

Mohammad Ali Ghaderi et al.[1] have proposed a Meta Search Engine to exploit social network data to improve web search results. The system modifies Meta Search Engine's multi agent based architecture by adding new agents to gather interaction data of users and process them to create user profiles based on the previous researches. These profiles are used to re-rank top search results of a web search engine and increase effectiveness of retrieval. Normalized Discounted Cumulative Gain (NDCG) measure is used to evaluate our system. Experimental results show the potential usefulness of social network data for improvement of web search effectiveness. Meta Search Engine will search a number of requests submitted to the members of the search engine and search pages to a certain degree of priority in accordance with the relationship between the order and display to the user.

Li Jianting [2] has described a Meta Search Engine retrieval results of collection process, such as the choice of the retrieval source; the rules of the retrieval, processing and retrieval results. Fuzzy integral algorithm uses various information sources to provide the right value information and decision making process to provide the necessary data, this type of information fusion to solve information retrieval and processing of uncertainty. Bernhard Krüpl and Robert Baumgartner [3] have proposed a Flight Meta Search Engine with Metamorph. They showed how data can be extracted from web forms to generate a graph of flight connections between cities. The flight connection graph allows us to vastly reduce the number of queries that the engine sends to airline websites in the most interesting search scenarios; those that involve the controversial practice of relative ticketing, in which agencies attempt to find lower price fares by using more than one airline for a journey. They described a system which attains data from a number of websites to identify promising routes and prune the search tree. Heuristics that make use of geographical information and an estimation of cost based on historical data are employed. The results are then made available to improve the quality of future search requests.

Felipe Bravo-Marquez et al.[4] have proposed a web services architecture for the retrieval of similar documents from the web. They focused on software engineering to support the manipulation of users' knowledge into the retrieval algorithm. An human evaluation for the relevance feedback of the system over a built set of documents is presented, showing that the proposed architecture can retrieve similar documents by using the main search engines. In particular, the document plagiarism detection task was evaluated, for which its main results are shown.

In [5] the idea of exploiting directly the scores of each search engine is proposed, where the main information is the relative rank of each result. Different ranking approaches are analyzed, for example Borda-fuse which is based on democratic voting, the Borda count or the weighted borda-fuse, in which search engines are not treated equally[6]. The document similarity retrieving problem has been studied by different researchers [7]. These approaches propose fingerprinting techniques for document representation into sets of relevant terms. Also, these approaches use Meta Search Engine architectures for retrieving an extended list of similar candidate documents. On the one hand, in document snippets are retrieved from search engines and compared with the query document using cosine similarity from their Vector Space Model.

III. Motivational Algorithms

Meta Search Engine is a system that provides unified access to multiple existing search engines. Now a day's study regarding Meta Search Engine and avoiding redundancy in Meta Search Engine has become more popular. Recently Ghaderi, M.A et al [1] have proposed a social network based

DOI: 01.IJIT.3.2.1194

Meta Search Engine. Their research introduced a Meta Search Engine, which exploits a social network to improve the web results. The system modifies Meta Search Engine's multi agent based architecture by adding new agents to gather interaction data of users and process them to create user profiles based on the previous researches. Hassan Sayyadi et al have introduced a clustering tool for optimizing the Meta Search Engine results. The method is known as NEws Meta Search REsult Clustering (NeSReC), which accepts queries directly from the user and collect the snippets of news which are retrieved by The AltaVista News Search Engine for the queries[18]. Afterwards, it performs the hierarchical clustering and labeling based on news snippets in a considerably tiny slot of time. These researches are motivated us in proposing a new method to improve the results of the Meta Search Engine[17].

The proposed Meta Search Engine is composed of multiple search engines classified from the deep search engine and the surface search engines. A deep web search engine is one, which retrieves information from the depth of the internet or from the invisible web. Normal surfing only gives the data from the surface of the internet, which is why it is called surface web search engines. The popular deep web search engines are Infomine, Incywincy, Complete Planet DeepPeep, etc and the popular surface search engines are Google, Yahoo, Bing, AltaVista and so on.

IV. Proposed Meta Search Engine Algorithms

The proposed method is concentrated mainly on two algorithms. The basic architecture is build up from the search results obtained from the deep web search and the surface web search. The user has full control on giving the query to the proposed Meta Search Engine. The input query is processed with the entire search engines. We have proposed two algorithms for the processing of the Meta Search Engine.

1. Algorithm-1: Concept similarity based Meta Search Engine
2. Algorithm-2: Cosine similarity based Meta Search Engine

Initially the query keyword given by the user is passed to the search engines, which are classified under the deep web search and surface web search. The search responses to the input query by giving set of documents, which satisfies the search criteria. The document consists of a number of keywords, which are the characteristics of the documents. The next process in the proposed method is a keyword based filtering.

A. Algorithm-1: Concept Similarity Based Meta Search Engine

The basic block diagram for concept similarity based Meta Search Engine is shown in the figure 1. A concept is a keyword which has some relation with the documents, and has some particular characteristics. This concept is related to documents and is also related with other concepts in the domain, which it belongs. The set of keywords are the input of this first stage of concept map extraction. Consider that we have a domain which consists of a set of concepts.

$$D = k_1, k_2, ..., k_n \tag{1}$$

In equation (1) D is the domain and k_i is concept that belongs to the domain. The aim of this step is to find the relation between the keywords and hence finding concepts to the domain. We adopt a sentence level windowing process, in which the window moves in a sliding manner. The text window formed is four term window which enclosed in a sentence. Initially, we find the highest frequent word and then, the approach finds the dependency of this word to other and other words to this.

$$freq(x) = \frac{X_n}{N_k}, X \in k. \tag{2}$$

Here, we find the most frequent element using equation (2). X_n represents the number of x present in the domain, where x is the element which is subjected for the frequency finding. N_k is the total number of elements in the domain. After finding the most frequent keyword, we have to find whether it belongs to a concept in the concept map. The selection that keyword to a concept is done by finding the inter relation between that keyword and other keywords. The bond between two keywords are obtained through finding the probability of occurrence of the keywords, we adopts a conditional probability for finding the relation between the keywords. The value of the dependency is used to extract the concept. If the keyword shows higher dependency between others, then it is considered as concept. Analysis of the method shows that the more the dependency the more the concept gets extracted from the text corpora. The dependency of the terms can be calculated through the following way,

$$dep(x:y) = \frac{P(x \mid y)}{P(y)}, x, y \in D. \tag{3}$$

$$P(x \mid y) = \frac{P(y \cap x)}{P(x)} \tag{4}$$

Here, the function $dep(x:y)$ is the function which is used for finding the dependency between the terms and thus extracting the concepts which is required for the concept map extraction using equation (3) and (4). The terms x and y represents the terms from the domain D. The function $P(.)$ is the probability of each word present in the domain. Here, we use both the conditional probability and the probability in the proposed approach. Thus concept belong to each document is found for the further processes. The concept is then considered as a term and the term belongs to the set of documents obtained after the search. The concepts is prominent character for the documents which posses it.

DOI: 01.IJIT.3.2.1194

The main advantage of the concept is, it is composed of one or more top N keywords extracted from the documents. The concepts possess prominent part in the proposed Meta Search Engine. The next phase of the proposed approach is the building an NxM matrix. In which we find the term frequency of the concept in the documents on accordance with the search query.

Example 1: if we have a query like *"data mining"*.

Steps.1 find the frequency of *"data"*, i.e. P(data)
Steps.2 find the frequency of *"mining"*, i.e. P(mining)
Steps.3 find the frequency of *"mining"* and *"data"*, i.e. P($\min ing \cap data$)

Steps.4 find $\frac{P(\min ing \cap data)}{P(\min ing)}$, i.e. $P(\min ing \mid data)$

Steps.5find $\frac{P(\min ing \mid data)}{P(data)}$, i.e. dep(mining : data)

Steps.6 dep values are passed to N x M formation.

In the similar way, concept in the every document is extracted and that terms are subjected for the N x M matrix calculation.

1) N x M matrix formation

The *dep* values of the document are arranged in an N x M matrix for the final calculation. In the N x M matrix the N is the number of concepts and M is the number of documents, which obtained from the search engines. All the dep values of the terms in the documents are calculated using the $dep(x:y) = \frac{P(x \mid y)}{P(y)}, x, y \in D.$ formulae. Then a row wise sum operation is initiated on each document to find its relevance to the search abased on the terms it posses.

$$\begin{array}{cccccc} & d_1 & d_2 & & d_n & \sum values \\ c_1 & dep(c_1,d_1) & dep(c_1,d_2) & & dep(c_1,d_n) & \sum_{n=1}^{N} dep(c_1,d_n) \\ & \ldots & \ldots & \ldots & \ldots & \ldots. \\ c_n & \ldots & \ldots & \ldots & \ldots & \sum_{n=1}^{N} dep(c_n,d_n) \end{array}$$

N x M matrix

Where d_1, d_2 …., $d_n \in D$, and $c_1, c_2, ..., c_n \in C$, C is the set concepts. The $\sum values$ are calculated and then it is sorted in descending order and a threshold is set for the $\sum values$. The those are higher than the thresholds are selected as the search results and those documents are retrieved to the user as the final search result.

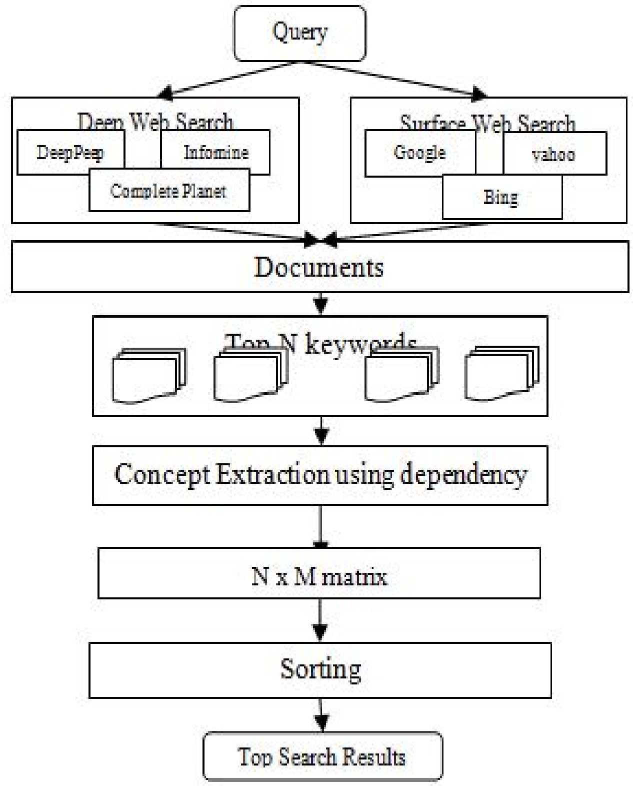

Figure 1. Block diagram for Algorithm 1

B. Algorithm-2 Cosine Similarity Based Meta Search Engine

The basic block diagram for consine similarity based Meta Search Engine is shown in the figure 2. The next phase of the proposed method deals with the term frequency (TF) and the inverse document frequency (IDF). The tf-idf function as the centre of the proposed method, i.e. their values controls the flow of the proposed method. The term frequency–inverse document frequency is a numerical statistic which reflects how important a word is to a document in a collection or corpus. It is often used as a weighting factor in information retrieval and text mining. The tf-idf value increases proportionally to the number of times a word appears in the document, but is offset by the frequency of the word in the corpus, which helps to control for the fact that some words are generally more common than others. Variations of the tf-idf weighting scheme are often used by search engines as a central tool in scoring and ranking a document's relevance given a user query. In the proposed method also we make use of the tf-idf through a specific formula.

The term count in the given document is simply the number of times a given term appears in that document. This count is usually normalized to prevent a bias towards longer documents to give a measure of the importance of the term *t* within the particular document *D*. Thus we have the term frequency,

$$TF(t,d) = No. of\, term\; t\, in\, document\, d$$

The inverse document frequency is a measure of whether the term is common or rare across all documents. It is ob

DOI: 01.IJIT.3.2.1194

ACEEE Int. J. on Information Technology, Vol. 3, No. 2, June 2013

tained by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of that quotient is given in equation (6).

$$IDF(t, D) = \log \frac{|D|}{|t \in d : d \in D|} \tag{6}$$

Where, $|D|$ is the cardinality of D, or the total number of documents in the corpus and the expression is the number of documents where the term *t* appears.

The proposed method formulates the tf-idf weightage with specific formulae, which can be given by below equation (7).

$$TF-IDF(t,d,D) = \frac{\sum_{t \in d} (TF \times IDF)_1 \times (TF \times IDF)_2}{\sum_{t \in d_1} \sqrt{(TF \times IDF)_1^2} \times \sum_{t \in d_2} \sqrt{(TF \times IDF)_2^2}}$$

This value of tf-idf is calculated for all the terms in the document and the resultant values are passed to processes the N x N matrix.

2)NxN Matrix Formation

The tf-idf values of the document are arranged in an N x N matrix for the final calculation. In the N x N matrix the N is the number of documents, which obtained from the search engines. All the tf-idf values of the terms in the documents are calculated using the *TF-IDF(t,d,D)* formulae. Then a row wise sum operation is initiated on each document to find its relevance to the search abased on the terms it posses.

$$\begin{array}{cccccc}
 & d_1 & d_2 & & d_n & \sum values \\
d_1 & [TF-IDF(t,d_1)+TF-IDF(t,d_1)] & [TF-IDF(t,d_1)+TF-IDF(t,d_2)] & & [TF-IDF(t,d_1)+TF-IDF(t,d_n)] & \sum_{n=1}^{N}[TF-IDF(t,d_1)+TF-IDF(t,d_n)] \\
 & \ldots & \ldots & \ldots & \ldots & \ldots. \\
d_n & \ldots & \ldots & \ldots & \ldots & \sum_{n=1}^{N}[TF-IDF(t,d_n)+TF-IDF(t,d_n)]
\end{array}$$

N x N matrix

Where d_1, d_2 …., d_n $\in D$. The $\sum values$ are calculated and then it is sorted in descending order and a threshold is set for the $\sum values$. The those are higher than the thresholds are selected as the search results and those documents are retrieved to the user as the final search result.

V. Results And Disussion

A. Testbed

The purpose of this work is to evaluate and compare different result merging algorithms under the context of Meta Search over the general purpose search engines. So we select 10 most popular general purpose search engines as the underlying component search engine. They are: Google, Bing, Infomine and IncyWincy. The reasons these search engines

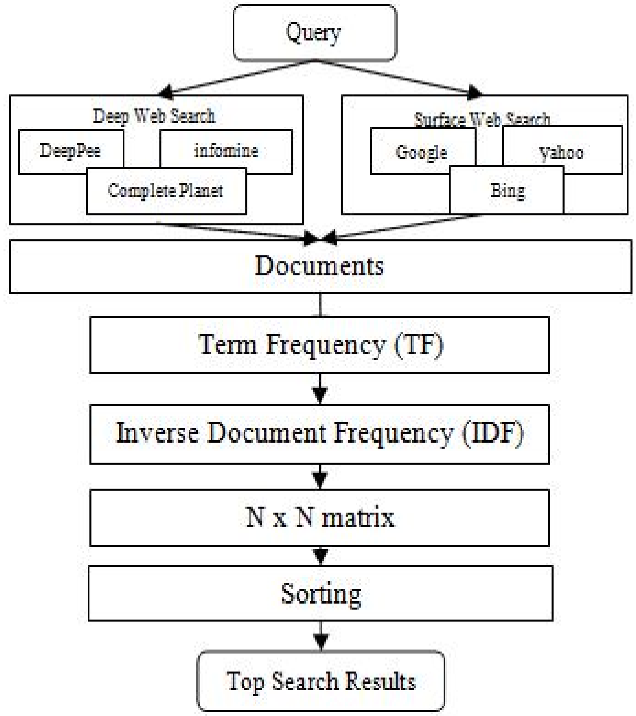

Figure 2. Block diagram for algorithm 2

are selected are: (1) they are used by nearly all the popular general purpose Meta Search Engines; (2) each of them has indexed a relatively large number of web pages; and (3) they adopt different ranking schemes. Even though we focus our work in the context of general purpose search engines, the result merging algorithms we proposed in this paper are completely independent of the search engine type. Each query is submitted to every component search engine. For each query and each search engine, the top 10 results on the first result page are collected. Information associated with each returned record is collected, including the URL, title, snippet and the local rank. Besides, the document itself is downloaded. The relevancy of each document is manually checked based on the criteria specified in the description and the narrative part of the corresponding TREC query. The collected data and the documents, together with the relevancy assessment result, form our testbed. The testbed is stored locally so it will not be affected by any subsequent changes from any component search engine.

Figure 3. GUI - Search

DOI: 01.IJIT.3.2.1194

ACEEE

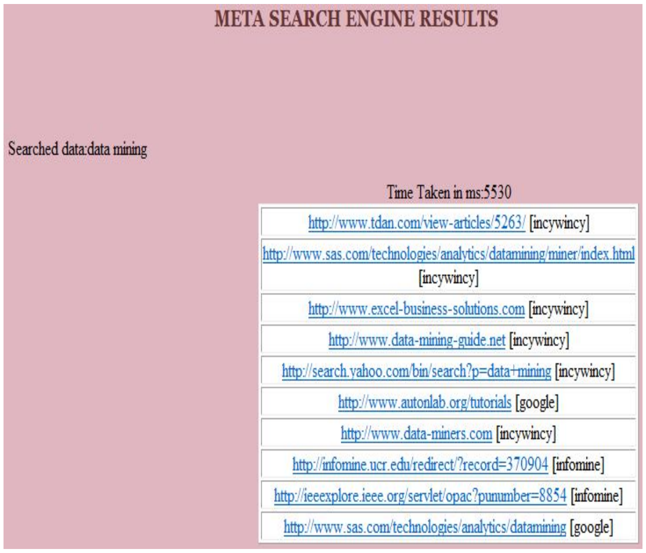

Figure 4. GUI- Results

B. Evaluation Function

Because it is difficult to know all the relevant documents to a query in a search engine, the traditional recall and precision for evaluating IR systems cannot be used for evaluating search/Meta Search Engines. A popular measure for evaluating the effectiveness of search engines is the TREC style average precision (TSAP). In this paper, TSAP at cutoff N, denoted as TSAP@N, will be used to evaluate the effectiveness of each merging algorithm using equation (8).

$$TSAP = (\sum_{i=1}^{N} r_i) / N \qquad (8)$$

Where $r_i = 1/i$ if the i[th] ranked result is relevant and $r_i = 0$ if the i[th] result is not relevant. It is easy to see that TSAP takes into consideration both the number of relevant documents in the top N results and the ranks of the relevant documents. TSAP tends to yield a larger value when more relevant documents appear in the top N results and when the relevant documents are ranked higher. For each merging algorithm, the average TSAP over all 50 queries is computed and is used to compare with other merging algorithms.

C. Performance Analysis

In this section we evaluate the performance of the proposed approach based on the different search engines and with our proposed Meta Search Engine. The evaluation is done for different search queries and their responses to the evaluation function. The performance of the proposed system will be different for different keywords given. The behavior of the search engines and the proposed method is evaluated according to the given keywords. In this process we consider the following search engines, Google and Bing as surface search engines and infomine and incywincy as deep web search engines. The GUI of the proposed Meta Search Engine in shown in figure 3. The performance of the above mentioned search engine are compared with the proposed Meta Search Engine based on the two algorithms, i.e. Concept based Meta Search Engine and IF-TDF weightage based Meta Search Engine.

TABLE I. ANALYSIS FACTORS

Search engines	Keywords
Google Bing Infomine Incywincy	Data Mining Network Security Data Replication Image Processing

Considering the analysis based on the keywords shown in the table 1. In the proposed method we have two algorithms to process with. So the data mining is given as input query to the search engines, according to the algorithm one it will generate some documents related to data and mining as shown in figure 4. Top n keywords from the documents are selected and then the concept is generated as “data mining” with the help of dependency value. Then the after the N x M matrix the relevant web sites or documents are selected. The responses of the keyword “data mining” is given below. We have plotted the TSAP values of the concept data mining in the below table.

TABLE II. TSAP VALUES FOR “DATA MINING”

Search Engine	N=10	N= 20
Google	0.40	0.55
Bing	0.30	0.35
Infomine	0.60	0.60
Incywincy	0.53	0.60
Proposed Algorithm 1	0.75	0.80
Proposed Algorithm 2	0.76	0.82

The analysis from the table 2 showed that the most ranked results are generated for the two proposed algorithms. The value obtained are 0.76 and 0.75 @N=10 and 0.82 and 0.80 @M=20 respectively for algorithm 2 and algorithm 1, which is higher value than the other search engines considered in the evaluation process. It can be stated that, the results are more feasible with the proposed method. Similarly all other keywords are processed with the above stated search engines.

Analysis based on keyword “Network Security”

TABLE III. TSAP VALUES FOR “NETWORK SECURITY”

Search Engine	N=10	N=20
Google	0.35	0.45
Bing	0.50	0.45
Infomine	0.55	0.50
Incywincy	0.63	0.65
Proposed Algorithm 1	0.78	0.81
Proposed Algorithm 2	0.76	0.83

The analysis from the table 3 showed that the most ranked results are generated for the two proposed algorithms. The value obtained are 0.76 and 0.78 @N=10 and 0.83 and 0.81 @M=20 respectively for algorithm 2 and algorithm 1, which is higher value than the other search engines considered in the evaluation process. The response to the second keyword is little bit higher than the first keyword.

Analysis based on keyword “Data replication”

DOI: 01.IJIT.3.2.1194

TABLE IV. TSAP VALUES FOR "DATA REPLICATION"

Search Engine	N=10	N=20
Google	0.40	0.45
Bing	0.38	0.43
Infomine	0.60	0.65
Incywincy	0.68	0.75
Proposed Algorithm 1	0.77	0.84
Proposed Algorithm 2	0.75	0.84

The analysis from the table 4 showed that the most ranked results are generated for the two proposed algorithms. The value obtained are 0.75 and 0.77 @N=10 and 0.84 and 0.84 @M=20 respectively for algorithm 2 and algorithm 1, which is higher value than the other search engines considered in the evaluation process. In this case, the Algorithm one performs little more sensitive to the give keyword than the other search measures.

Analysis based on keyword "Image Processing"

TABLE V. TSAP VALUES FOR "IMAGE PROCESSING"

Search Engine	N=10	N=20
Google	0.57	0.59
Bing	0.48	0.53
Infomine	0.70	0.75
Incywincy	0.78	0.85
Proposed Algorithm 1	0.74	0.82
Proposed Algorithm 2	0.69	0.80

The analysis from the table 5 showed that the most ranked results are generated for the two proposed algorithms. The value obtained are 0.69 and 0.74 @N=10 and 0.80 and 0.82 @M=20 respectively for algorithm 2 and algorithm 1, which is higher value than the other search engines considered in the evaluation process.

The evaluation of the four key words states that our search engine is sensitive to the user input and it has upper hand over the other methods in most of the other methods in different search criteria.

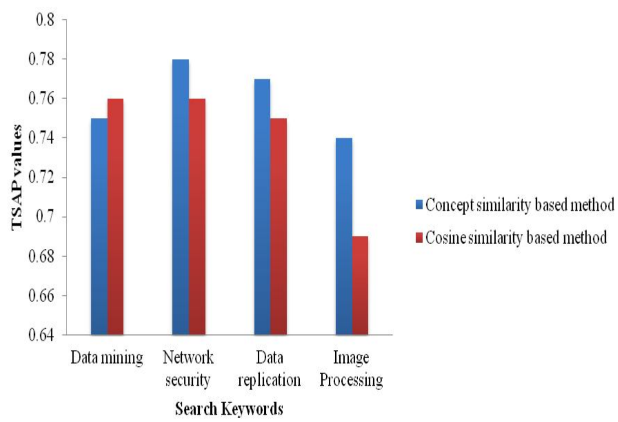

Figure 5. TSAP value Comparison 1

In Figure 5, we have plotted the comparison of the TSAP values of different keywords with respect to the concept similarity based algorithm and the cosine similarity based algorithm. The figure shows that the cosine similarity algorithm performs little less as compared to the concept similarity based algorithm. Even though, by neglecting their individual performance the proposed algorithm performs a way higher than the traditional search engines.

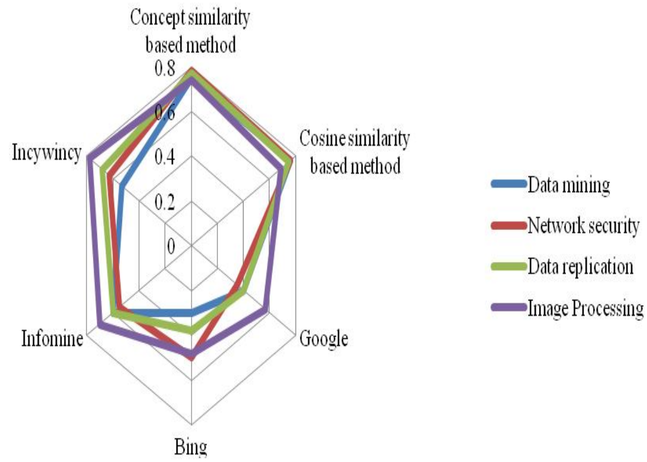

Figure 6. TSAP value comparison 2

The plotting in figure 6 shows the performance of the proposed approach with all other search engines considered in the experiment.

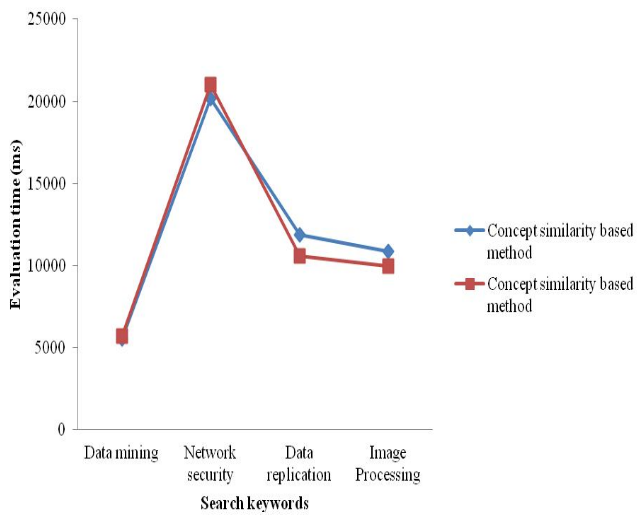

Figure 7. Execution time

The figure 7 shows the time required for searching the results from the web in relevant to the search keyword. The plotting represents the time taken for the execution of algorithms regarding the searching of the keywords. Both the algorithm has taken almost same time for delivering the search results.

D. Comparative Analysis

In this section, a comparison of the proposed approach has been plotted with an existing Meta Search algorithm. The existing Meta Search is used for evaluation of result merging strategies for Meta Search Engines. The above stated approach implemented three algorithms for the evaluating the Meta Search process and the algorithms are derived based on the similarity of the documents and the similarity measures are SRRsim, SRRrank and SRRsimF. The three algorithms are evaluated with the proposed Meta Search Engine algorithm with evaluation criteria TSAP. The algorithm cosine similarity

measure and concept similarity measure is compared with SRRsim, SRRrank and SRRsimF with different N values. The responses from the comparison study are plotted in the following graph.

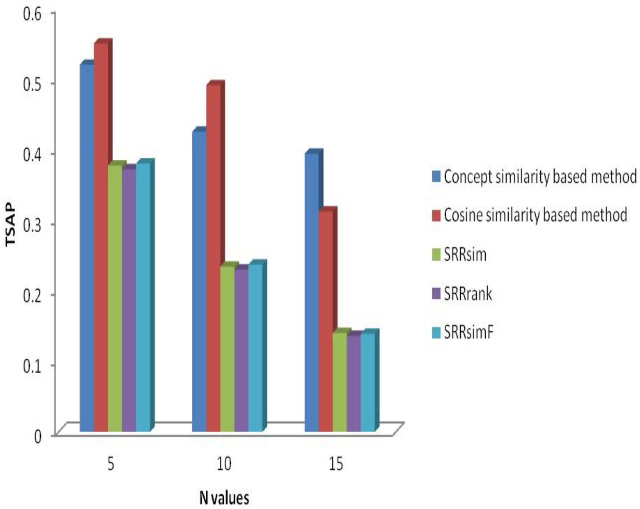

Figure 8.Comparison Analysis

The figure 8 shows the comparative study of the proposed concept similarity and cosine similarity algorithm with SRRsim, SRRrank and SRRsimF. The analysis shown that the proposed cosine and concept algorithm are performs better than the existing three algorithms. The responses of every algorithm towards the N values are proportional, i.e. as the N value increases the TSAP value decreases accordingly. The maximum response of TSAP values obtained is 0.55 and is which the result obtained by the cosine similarity. On comparison cosine similarity algorithm proves better response from the others.

Conclusions

The search engine has been replaced with Meta Search Engines now days for getting more accurate and precise outputs. The Meta Search Engines are used because they are capable of overcome the limitations faced by the normal search engines. The proposed method introduces two algorithms, which improves the Meta Search Engine results. The proposed method defines two algorithms, they are concept similarity based method and cosine similarity based method. The first one considers the keyword as a concept and find its relevance to the search criteria, on the other hand, cosine similarity make use of the term frequency and inverse document frequency. The experimental results have shown that the proposed algorithm out performs the other search engines. The evaluation criteria we used in the proposed algorithms TSAP. The futuristic advancements can be done by incorporating different evaluation parameters to the proposed methods.

References

[1] Mohammad Ali Ghaderi, Nasser Yazdani and Behzad Moshiri, " A Social Network based Meta Search Engine", International Symposium on Telecommunications, pp: 744-749, 2010.

[2] Li Jianting, "Processing and fusion of meta search engine retrieval results ", International Conference on Electric Information and Control Engineering, pp: 3442-3445, 2011.

[3] Bernhard Krüpl and Robert Baumgartner, " Flight Meta Search Engine with Metamorph", Transactions on ACM journlas, pp:1069- 1070, 2009.

[4] Felipe Bravo-Marquez, Gaston L'Huillier, Sebasti´an A. R´ios, Juan D. Vel´asquez and Luis A. Guerrero, "DOCODE-Lite: A Meta Search Engine for Document Similarity Retrieval", In Proceedings of the 14th international conference on Knowledge based and intelligent information and engineering systems, pp: 93-102, 2010.

[5] Rasolofo, Y., Abbaci, F., Savoy, J., "Approaches to collection selection and results merging for distributed information retrieval". In Proceedings of the Tenth International Conference on Information and Knowledge Management, pp.191–198, 2001.

[6] Aslam, J.A., Montague, M, "Models for metasearch", In Proceedings of the 24th Aannual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 276–284. 2001.

[7] Zaka, B, "Empowering plagiarism detection with a web services enabled collaborative network", Journal of Information Science and Engineering, vol 25, 1391–1403, 2009.

[8] Selberg, E., Etzioni, O, "The metacrawler architecture for resource aggregation on the web", IEEE Expert journal,pp: 11–14, 1997.

[9] Bernard J. Jansen, Amanda Spink, Sherry Koshman, "Web Searcher Interaction With the Dogpile.com Metasearch Engine", JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE AND TECHNOLOGY, Vol. 58, pp: 744-755, 2007.

[10] S. Gauch, G. Wang, M. Gomez, "ProFusion:Intelligent Fusion from Multiple, Distributed Search Engines", Journal of Universal Computer Science, vol. 2, 1996.

[11] A. E. Howe and D. Dreilinger, "SavvySearch: A meta search engine that learns which search engines to query", AI Magazine, vol. 18, 1997.

[12] A. H. Keyhanipour, B. Moshiri, M. Piroozmand, C. Lucas, "WebFusion: Fundamentals and Principals of a Novel Meta Search Engine". IJCNN'2006, pp.4126-4131, 2006.

[13] B. Smyth, J. Freyne, M. Coyle, P. Briggs, E. Balfe, and T. Building, "I-spy: anonymous, community based personalization by collaborative web search," in In Proceedings of the 23rd SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence, pp. 367-380, 2003.

[14] Chignell, M.H., Gwizdka, J., & Bodner, R.C, "Discriminating meta search: A framework for evaluation", Information Processing and Management,vol. 35, 337–362, 1999.

[15] Selberg, E., & Etzioni, O, "The MetaCrawler architecture for resource aggregation on the Web", IEEE Expert, Vol 12, 11–14, 1997.

[16] Meng, W., Yu, C., & Lui, K.-L, "Building efficient and effective meta search engines". ACM Computing Surveys, vol. 34, pp: 48–89, March 2002.

[17] Spink, A., Lawrence, S., & Giles, L. "Inquirus Web meta search tool: Auser evaluation", In Proceedings of WebNet 2000, pp. 819–820, 2000.

[18] Hassan Sayyadi, Sara Salehi, Hassan AbolHassani, "NeSReC: A News Meta Search Engines Result Clustering Tool", Advances and innovations in systems, 2006.

DOI: 01.IJIT.3.2.1194